REMARKS/ARGUMENTS

Claims 1-20, 27, and 31-46 are pending in the application. Claims 1, 16, 33, 35, 38, and 42-46 are amended herein. The Applicant hereby requests further examination and reconsideration of the application in view of the foregoing amendments and these remarks.

On 12/17/04, the Examiner participated in a telephonic interview with the Applicant's attorney Steve Mendelsohn. The Applicant thanks the Examiner for the courtesy of that interview.

Claim Objections

In paragraph 2 of the office action, the Examiner objected to claims 38-46 because of certain informalities. In response, the Applicant has amended claims 38, 42-43, and 45-46 as suggested by the Examiner. The Applicant submits that none of these amendments have been made to overcome any prior art rejections.

Rejections under 35 U.S.C. 112, First Paragraph

In paragraph 4, the Examiner rejected claims "1-15 and 16-32" under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement. (The Applicant notes that claims 21-26 and 28-30 were previously canceled and are no longer pending.) In particular, the Examiner stated that "the disclosure of the drawings" does not teach the generation of "two or more sum outputs." It is true that the figures show an adder that generates a single sum output at a time. For example, Fig. 2 shows adder 70, which generates sum 72. However, over time, as different sets of data are shifted into shift registers 64_1-64_4 and as different sets of coefficient data are loaded into tap changer 76, adder 70 will generate two or more different sum outputs.

According to claims 1 and 16, two or more sum outputs are generated between consecutive shiftings of new data into the shift registers. In other words, in the exemplary context of Fig. 2, after shifting new data into shift registers 64_1-64_4, two or more different sum outputs are generated before any other new data is shifted into the shift registers. That does not mean that two or more different sum outputs are generated at the same time. Rather, two or more different sum outputs are generated one after the other.

Continuing with the example of Fig. 2, the coefficients stored in tap changer 76 are shifted at a rate (e.g., four times) faster than the rate at which new data is shifted into shift registers 64_1-64_4. See, e.g., page 6, lines 24-26, and page 7, lines 5-11, of the specification. As such, in this exemplary implementation, assuming that the shift registers are currently filled with data, the flow of processing within the digital filter of Fig. 2 may be summarized as follows:

Step (1):	Shift new data into shift registers 64_1-64_4, store coefficients in tap changer 76, and generate 1st sum output.
Step (2):	Shift coefficients within tap changer 76 and generate 2nd sum output.
Step (3):	Shift coefficients within tap changer 76 and generate 3rd sum output.
Step (4):	Shift coefficients within tap changer 76 and generate 4th sum output.
Step (5):	Shift new data into shift registers 64_1-64_4, re-store coefficients in tap changer 76, and generate 5th sum output.
Step (6):	Shift coefficients within tap changer 76 and generate 6th sum output.
Step (7):	Shift coefficients within tap changer 76 and generate 7th sum output.
Step (8):	Shift coefficients within tap changer 76 and generate 8th sum output.

Step (9): Shift new data into shift registers 64_1-64_4, re-store coefficients in tap changer 76, and generate 9th sum output.

...and so on...

This is an example of what is meant in claim 1 by the recitation that "two or more sum outputs are generated between consecutive shiftings of new data into the shift registers." In particular, new data is shifted into the shift registers only at Steps (1), (5), and (9), while a different sum output is generated at every step. Thus, for example, between the consecutive shiftings of new data into the shift registers of Steps (1) and (5), four different sum outputs are generated (i.e., the 1st, 2nd, 3rd, and 4th sum outputs). Similarly, between the consecutive shiftings of new data into the shift registers of Steps (5) and (9), four more, different sum outputs are generated (i.e., the 5th, 6th, 7th, and 8th sum outputs).

In view of the foregoing, the Applicant submits that the generation of "two or more sum outputs" is in fact enabled by the present application. The Applicant submits therefore that the rejection of claims under 35 U.S.C. 112, first paragraph, has been overcome.

Rejections under 35 U.S.C. 112, Second Paragraph

In paragraph 6, the Examiner rejected claims "1-32 and 38-46" under 35 U.S.C. 112, second paragraph, as being indefinite. (The Applicant notes again that claims 21-26 and 28-30 were previously canceled and are no longer pending.) In response, the Applicant has amended the claims as follows:

- Claims 1 and 16 have been amended to recite "the at least two multiple stage shift registers" instead of "the shift registers."

- Claim 44 has been amended to recite "the *N* multiple-stage shift registers" instead of "the shift registers."

None of these amendments have been made to overcome any prior art rejections.

Regarding claims 38 and 46, the Examiner stated that "each set of data" lacks antecedent basis. First of all, the Applicant submits that the term "each set of data" is an indefinite phrase, similar to "a set of data" or "one set of data." A corresponding definite phrase would be "said each set of data," similar to "said one set of data" being a definite phrase corresponding to the indefinite phrase "one set of data." Indefinite phrases do not require antecedent basis.

Moreover, the phrase "each set of data" does in fact have implicit antecedence in the recitation of "a datum from a corresponding stage of a corresponding shift register" at lines 4-5. Since there are *N* shift registers ($N>1$), each having multiple stages, as recited at line 2, it is inherent that the *N* multiple-stage shift registers are adapted to store a set of data. Furthermore, the nature of shift registers is that data is shifted through them over time. Thus, at different times, the shift registers store different sets of data. The recitation of "each set of data stored in the shift registers" refers individually to different sets of data stored in the shift registers at different times.

As such, the Applicant submits that "each set of data" in claims 38 and 46 does not lack antecedent basis.

In view of the foregoing, the Applicant submits that the rejections of claims under 35 U.S.C. 112, second paragraph, have been overcome.

Prior Art Rejections

In paragraph 8, the Examiner rejected claims 1, 4-5, 12, 16, 19-20, 27, 33, 35-36, and 38-46 under 35 U.S.C. 102(e) as being anticipated by Zhou. In paragraph 10, the Examiner rejected claims 2-3, 17-18, 34, and 37 under 35 U.S.C. 103(a) as being unpatentable over Zhou in view of Nishida. In paragraph 11, the Examiner rejected claims 6-11 and 32 under 103(a) as being unpatentable over Zhou in view of Nishida and further in view of Schilling. In paragraph 12, the Examiner rejected claims 13-14 and 31 under 103(a) as being unpatentable over Zhou in view of Nishida and further in view of Black. In paragraph 13, the Examiner rejected claim 15 under 103(a) as being unpatentable over Zhou in view of Nishida and further in view of Gronemeyer. For the following reasons, the Applicant submits that all of the now-pending claims are allowable over the cited references.

Claims 1 and 16

According to claims 1 and 16, two or more sum outputs are generated between consecutive shiftings of new data into the shift registers. This feature relates to a digital filter that generates two or more different sums for each set of data stored in the shift registers.

Zhou teaches, in Fig. 3 and on column 3, line 28, to column 4, line 24, a matched filter that has two data register sequences (R11 to R1n and R21 to R2n). The output of each data register R1i and the output of its corresponding data register R2i are input to a corresponding selector SELi, which selectively outputs one of those two data register outputs to a corresponding exclusive-or-gate XORi, which combines the selected data register output with a corresponding bit of the PN code sequence stored in shift register SREG. The outputs from the n XOR gates are then combined in a current addition circuit ADD to generate the analog output signal A_{out}.

In Zhou, an analog data signal Ain is digitized and then double-sampled when loaded into the two data register sequences, where "one and only one of the data registers holds the output of the A/D converter at one time." After a datum is loaded into the last register R1n, the next datum for that data register sequence is loaded into the first register R11 (overwriting the previously stored datum value), the following datum for that data register is loaded into the second register R12, and so on. As such, at any given time, the "oldest" datum can be stored in any of the data registers (e.g., R1j) with the "newest" datum stored in the previous data register in the sequence (e.g., R1 j-1), where "j-1" is n, for the particular case when "j" is 1.

To match this data-loading cycle, the PN code sequence in Zhou is circularly shifted in shift register SREG to ensure that the proper PN code bits are combined at the XOR gates with the corresponding data values in the selected data registers. According to Zhou, the clock CLKS, which controls the cycling of the PN code sequence in shift register SREG, is "synchronous with" the clocks CLK0 and CLK1, which control the loading of data into the two data register sequences, "such that the PN code sequence is shifted and circulated corresponding to the data input to the data registers from the A/D converter." (The Applicant assumes that, to conform to Fig. 3 and earlier passages in column 3, "CLK1 and CLK2" at line 67 of column 3 of Zhou should be "CLK0 and CLK1.") Thus, in Zhou, there is exactly one value of the analog output signal A_{out} generated for each set of data stored in the selected data registers.

In particular, assuming that the data registers are currently filled with data, the flow of processing of Fig. 3 in Zhou may be summarized as follows:

Step (1): Store output of A/D into data register R11, shift PN code bits within shift register SREG, select data registers R11 through R1n, and generate 1st output value.

Step (2): Store output of A/D into data register R21, select data registers R21 through R2n, and generate 2nd output value.

Step (3): Store output of A/D into data register R12, shift PN code bits within shift register SREG, select data registers R11 through R1n, and generate 3rd output value.

Step (4): Store output of A/D into data register R22, select data registers R21 through R2n, and generate 4th output value.

Step (5): Store output of A/D into data register R13, shift PN code bits within shift register SREG, select data registers R11 through R1n, and generate 5th output value.

Step (6): Store output of A/D into data register R23, select data registers R21 through R2n, and generate 6th output value.

Step (7): Store output of A/D into data register R14, shift PN code bits within shift register SREG, select data registers R11 through R1n, and generate 7th output value.

Step (8): Store output of A/D into data register R24, select data registers R21 through R2n, and generate 8th output value.

Step (9): Store output of A/D into data register R15, shift PN code bits within shift register SREG, select data registers R11 through R1n, and generate 9th output value.

... and so on ...

As indicated by this summary, every time another bit of data is stored into a data register, one and only one output value is generated. In other words, in Zhou, no more than one output value is generated between consecutive storings of new data into the data register sequences. This is different from the invention of claims 1 and 16, where two or more sum outputs are generated between consecutive shiftings of new data into the shift registers.

While it is true, as the Examiner stated on page 5 of the office action, that Zhou teaches "at the CLK1 and CLK2 the ADD provides at least two sums," those two sums correspond to two different storings of data into shift registers SFREG1 and SFREG2 of Fig. 23. In particular, during a cycle of CLK1, new data is shifted from the A/D into SFREG1 and a first sum is generated. Then, during the very next cycle of CLK2, data is shifted from the A/D into SFREG2 and a second sum is generated. Again, this is different from the recitation in claims 1 and 16 that "two or more sum outputs are generated between consecutive shiftings of new data into the shift registers."

None of the other cited references of record teach the features of claims 1 and 16 that are missing from Zhou.

For all these reasons, the Applicant submits that claims 1 and 16 are allowable over the cited references. Since claims 2-15 and 17-32 depend variously from claims 1 and 16, it is further submitted that those claims are also allowable over the cited references.

Claim 33

According to the method of claim 33, (a) digital data is shifted into first and second multiple stage shift registers, (b) an output from each stage of the first and second multiple stage shift registers is

multiplied by an associated, respective tap weight to produce a plurality of products; (c) the plurality of products are combined to form a sum, (d) the tap weights are circularly shifted, and (e) steps (b) and (c) are repeated at least once before step (a) is repeated. In light of the previous discussion, the invention of claim 33 differs from the teachings in Zhou for (at least) the following reason.

In Zhou, at most a single output is generated for each loading of new data into the shift register sequences. Thus, Zhou does not teach the generation of a second sum before shifting new data into first and second shift registers.

In rejecting claim 33, the Examiner stated, on page 5, that CLK1 "is shifted by half a chip time from CLK0 at least once before loading new data." The Applicant respectfully submits that the Examiner's statement does not make sense. CLK0 and CLK1 run at the same rate, but are shifted in time by half a chip relative to one another. In other words, the two clocks are 180-degrees out of phase with respect to one another. When CLK0 rises, CLK1 falls, and when CLK0 falls, CLK1 rises. This is exactly what enables the circuitry in Zhou to oversample the input data.

Referring to Zhou's Fig. 23, first SFREG1 receives a bit of input data based on, e.g., a rising edge of CLK1, then SFREG2 receives the next bit of input data based on, e.g., the next rising edge of CLK2 (i.e., half a chip after the rising edge of CLK1), then SFREG1 receives the next bit of input data based on the next rising edge of CLK1, then SFREG2 receives the next bit of input data based on the next rising edge of CLK2, and so on. Significantly, in Zhou, at every oversampling of the input data, a single new output value is generated.

None of the other cited references of record teach these features of claim 33 that are missing from Zhou.

For all these reasons, the Applicant submits that claim 33 is allowable over the cited references. Since claim 34 depends from claim 33, it is further submitted that claim 34 is also allowable over the cited references.

Claim 35

According to the method of claim 35, (a) data is shifted into N multiple stage shift registers, each of the N multiple stage shift registers having at least L stages, N and L being integers, N being at least 2, (b) an output from each of the at least L stages of the N multiple stage shift registers is multiplied by a corresponding tap weight to produce a plurality of products, (c) the plurality of products are combined to form a sum, (d) the tap weights are circularly shifted, (e) steps b, c, and d are repeated N-2 times before step a is repeated, and (f) steps b and c are repeated again before step a is repeated. In light of the earlier discussion, the invention of claim 35 differs from the teachings in Zhou for (at least) the following reason.

In Zhou, at most a single output is generated for each loading of new data into the shift register sequences. Thus, Zhou does not teach the generation of a second sum before shifting new data into N shift registers.

None of the other cited references of record teach these features of claim 35 that are missing from Zhou.

For all these reasons, the Applicant submits that claim 35 is allowable over the cited references. Since claims 36-37 depends from claim 35, it is further submitted that those claims are also allowable over the cited references.

Claims 38 and 46

According to claims 38 and 46, the digital filter has *N* multiple-stage shift registers, a tap changer, a plurality of multiplying elements, and an adder, where the digital filter is adapted to generate two or more sums for each set of data stored in the shift registers. For at least some of the same reasons provided earlier, the Applicant submits that claims 38 and 46 are allowable over the cited references. Since claims 39-45 depend variously from claim 38, it is further submitted that those claims are also allowable over the cited references.

Claim 44

According to claim 44, the *N* multiple-stage shift registers do not all have the same number of stages. The Applicant submits that, in Zhou, all of the data registers have the same number of stages. For example, in Fig. 3, each of the two data registers has *n* stages. The same is true for the two shift registers SFREG1 and SFREG2 of Fig. 23. Zhou does not teach or even suggest an implementation in which shift registers do not all have the same number of stages. As such, the Applicant submits that this provides additional reasons for the allowability of claim 44 over Zhou.

In view of the foregoing, the Applicant submits that the rejections of claims under Sections 102(e) and 103(a) have been overcome.

In view of the above amendments and remarks, the Applicant believes that the now-pending claims are in condition for allowance. Therefore, the Applicant believes that the entire application is now in condition for allowance, and early and favorable action is respectfully solicited.

Respectfully submitted,

Date: 2/16/05

Customer No. 22186
Mendelsohn & Associates, P.C.
1515 Market Street, Suite 715
Philadelphia, Pennsylvania 19102

Steve Mendelsohn
Registration No. 35,951
Attorney for Applicant
(215) 557-6657 (phone)
(215) 557-8477 (fax)